

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization  
International Bureau



(43) International Publication Date  
12 June 2003 (12.06.2003)

PCT

(10) International Publication Number  
WO 03/048794 A1

- (51) International Patent Classification<sup>7</sup>: G01R 31/28      (81) Designated States (*national*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZM, ZW.
- (21) International Application Number: PCT/US02/36246
- (22) International Filing Date:  
12 November 2002 (12.11.2002)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:  
60/336,586      4 December 2001 (04.12.2001) US  
10/142,556      10 May 2002 (10.05.2002) US
- (71) Applicant: INTELLITECH CORPORATION [US/US]; 70 Main Street, Durham, NH 03824 (US).
- (72) Inventors: RICCHETTI, Michael; 54 Cathedral Circle, Nashua, NH 03063 (US). CLARK, Christopher, J.; 22B Cedar Point Road, Durham, NH 03824 (US).
- (74) Agents: LEBOVICI, Victor, B. et al.; Weingarten, Schurigin, Gagnebin & Lebovici, LLP, Ten Post Office Square, Boston, MA 02109 (US).
- (84) Designated States (*regional*): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

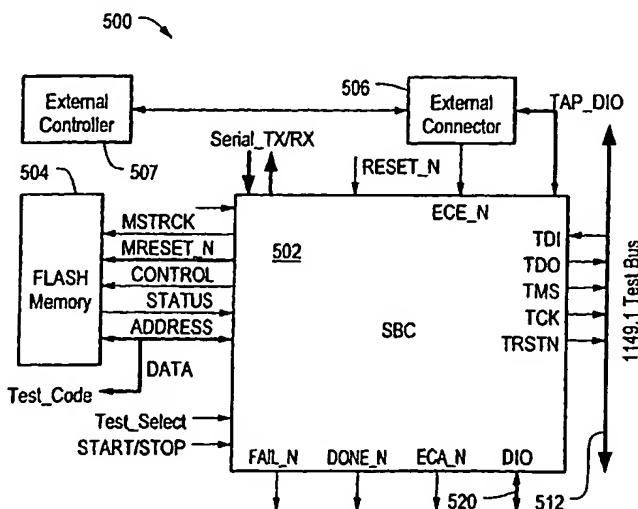
- as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii)) for all designations
- as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii)) for all designations

[Continued on next page]

(54) Title: METHOD AND APPARATUS FOR EMBEDDED BUILT-IN SELF-TEST (BIST) OF ELECTRONIC CIRCUITS AND SYSTEMS



WO 03/048794 A1



(57) Abstract: An embedded electronic system built-in self-test controller architecture that facilitates testing and debugging of electronic circuits and in-system configuration of programmable devices. The system BIST controller architecture includes an embedded system BIST controller, an embedded memory circuit, an embedded IEEE 1149.1 bus, and an external controller connector. The system BIST controller is coupled to the memory circuit and the IEEE 1149.1 bus, and coupleable to an external test controller via the external controller connector. The external test controller can communicate over the IEEE 1149.1 bus to program the memory and/or the system BIST controller circuitry, thereby enabling scan vectors to be debugged by the external test controller and then downloaded into the memory for subsequent application to a unit under test by the system BIST controller.

BEST AVAILABLE COPY



**Published:**

— *with international search report*

*For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.*

5

## TITLE OF THE INVENTION

METHOD AND APPARATUS FOR EMBEDDED BUILT-IN SELF-TEST (BIST) OF  
ELECTRONIC CIRCUITS AND SYSTEMS

## 10 CROSS REFERENCE TO RELATED APPLICATIONS

This application claims priority of U.S. Provisional Patent Application No. 60/336,586 filed December 4, 2001 entitled METHOD AND APPARATUS FOR EMBEDDED BUILT-IN SELF-TEST (BIST) OF ELECTRONIC CIRCUITS AND SYSTEMS.

15

STATEMENT REGARDING FEDERALLY SPONSORED RESEARCH OR  
DEVELOPMENT

N/A

## 20 BACKGROUND OF THE INVENTION

The present invention relates generally to Built-In Self-Test (BIST) of Integrated Circuits (ICs), Printed Circuit Boards (PCBs), and systems, and more specifically to an apparatus and method for embedding BIST capability within ICs, PCBs, and 25 systems.

Techniques are known that employ scan testing for providing manufacturing test, debug, and programming of electronic circuits. Such scan testing techniques are often performed according to the IEEE 1149.1 Standard Test Access Port and 30 Boundary Scan Architecture specification ("the IEEE 1149.1 Standard"), which is incorporated herein by reference. The IEEE 1149.1 Standard may also be used to provide In-System Configuration (ISC) of programmable circuits.

The IEEE 1149.1 Standard was initially developed for interconnect testing of PCBs. The IEEE 1149.1 Standard employs a boundary scan path to facilitate access to Input/Output (I/O) pins of devices mounted on a PCB. In addition, the IEEE 1149.1 Standard may be used to access internal scan paths of an IC to facilitate test, debug, ISC, or programming of ICs, PCBs, and systems.

Fig. 1 depicts the conventional IEEE 1149.1 Standard Architecture 100. As shown in Fig. 1, an IC compliant with the IEEE 1149.1 Standard has four (optionally, five) additional component pins TDI, TDO, TCK, and TMS (optionally, TRSTN), which form a Test Access Port (TAP). The IEEE 1149.1 Standard facilitates the connection of TAP ports of multiple electronic circuits to form an IEEE 1149.1 bus, thereby allowing the connected circuits to be accessed using a common TAP protocol. This is typically achieved by connecting the serial data pins TDI and TDO of individual devices in a daisy chain fashion such that the TDO output from a previous device along the chain is connected to the TDI input of a next device in the chain. Then, by connecting all of the TMS and TCK (optionally TRSTN) pins of the devices in common, an overall TAP bus is formed.

Fig. 2 depicts a conventional IEEE 1149.1 bus in a daisy chained configuration 200. As shown in Fig. 2, the TDI pin on a first device U1 202.1 and the TDO pin on a last device Un 202.n are used as the serial data input and serial data output of the IEEE 1149.1 bus, respectively. The bus configuration 200 shown in Fig. 2 is typically employed on a single PCB.

Fig. 3 depicts a conventional IEEE 1149.1 bus in a multi-drop configuration 300. When utilized within a system of PCBs, the multi-drop configuration 300 provides for a single TAP bus across a backplane of the system and allows each PCB to make connections to the same set of wires on the multi-drop bus. Because TCK, TMS, TDI and TRSTN are input signals, these signals

can be directly connected across the system backplane to each of the TAPs of the individual PCBs. However, signal clashes may result when connecting the multiple TDO outputs onto the single TDO wire of the multi-drop bus. To avoid such signal clashes,  
5 the IEEE 1149.1 Standard requires that the TDO output drive out only when serial data is being shifted into or out of the TAP's TDI and TDO pins. Accordingly, such serial-shift is controlled by internal states of the TAP Controller so that the TDO drive is enabled only during the Shift-IR or the Shift-DR states of  
10 the TAP Finite State Machine (FSM). At all other times, the TDO output is disabled by forcing it into an inactive or high-impedance state. Either a specialized version of the TAP controller or an Addressable TAP Linking (ATL) circuit may be employed to implement the multi-drop bus configuration 300 of  
15 Fig. 3. Such an ATL circuit is described in co-pending U.S. Patent Application No. 60/303,052 filed July, 5, 2001 entitled METHOD AND APPARATUS FOR OPTIMIZED PARALLEL TESTING AND ACCESS OF ELECTRONIC CIRCUITS.

An external test controller can be connected to the TDI,  
20 TDO, TMS, TCK, and TRSTN lines of the respective IEEE 1149.1 bus in the daisy chained or multi-drop configurations 200 and 300. The external test controller can then communicate with the respective Units Under Test (UUTs) 202.1-202.n or 302.1-302.n using the IEEE 1149.1 bus protocol. These bus configurations  
25 200 and 300 are commonly used in production manufacturing of electronic systems, in which the external test controller is typically some form of Automatic Test Equipment (ATE) such as an In-Circuit Tester (ICT) or a Personal Computer (PC) based boundary scan tool.

30 There is a need for embedding Built-In Self-Test (BIST) capability within a system to be tested, on one or more PCBs of the system to be tested, and/or on one or more ICs of the system to be tested. Such embedded BIST capability would allow

circuitry resident within the system to apply scan vector sequences that would otherwise be applied by an external test controller. Further, such embedded BIST capability would enable tests to be readily performed either remotely or in the field.

5 For example, such testing may be performed automatically at system power-up or by invoking the embedded BIST circuitry at some other time.

Fig. 4 depicts a conventional configuration 400 for providing embedded BIST capability in electronic systems. The 10 BIST configuration 400 comprises an architecture designed around a general-purpose microprocessor 402 and a data conversion circuit 404 that converts between the parallel data/protocol of the microprocessor 402 and the serial scan protocol (e.g., the IEEE 1149.1 protocol) of the UUT. As shown in Fig. 4, both 15 Read-Only Memory (ROM) 406 and Random Access Memory (RAM) 408 are connected to a bus 410 of the microprocessor 402. The ROM 406 stores program code and the RAM 408 stores data used when the microprocessor 402 executes the stored programs.

The embedded BIST configuration 400 further includes 20 interface logic 412 connected between the microprocessor 402 and the parallel/serial protocol converter circuit 404 to match the address and control signals of the microprocessor 402 to those of the converter circuit 404. The parallel protocol/data of the microprocessor 402 are converted to the inputs and outputs 25 forming the IEEE 1149.1 bus 414 by the parallel/serial converter 404. The IEEE 1149.1 bus 414 may then be employed to drive a respective IEEE 1149.1 bus on a PCB or within an IEEE 1149.1 bus configuration such as the daisy chained configuration 200 (see Fig. 2) or the multi-drop configuration 300 (see Fig. 3). 30 Moreover, the embedded BIST configuration 400 includes an external connector 416 that bypasses the parallel/serial protocol converter circuit 404 and enables an external test controller 407 to be connected in place of the data conversion

circuit 404. When the external test controller 407 is connected in this manner, an OE signal operates to disable the parallel/serial protocol converter circuit 404 from controlling the IEEE 1149.1 bus 414, thereby allowing the external test 5 controller 407 to control the IEEE 1149.1 bus 414.

As described above, the conventional embedded BIST configuration 400 includes the microprocessor 402, non-volatile storage in the form of the ROM 406, and read/write storage in the form of the RAM 408. For such a microprocessor-based 10 approach, a user (i.e., a human operator) normally writes program code (e.g., C code) and compiles and links the code with a library of scan test functions. The linked code resides in the ROM 406 (which also includes the scan vector data) and is executed by the microprocessor 402 to apply and evaluate the 15 scan vectors. The RAM 408 is employed for temporary storage during various microprocessor operations such as comparing actual scan-out data with expected scan-out data.

As a result, the conventional configuration 400 for providing embedded BIST capability requires customization of the 20 embedded test solution for each application. Specifically, the program code is developed and debugged for each specific system. In addition, the scan vectors that are normally applied by the external test controller are converted to operate with the embedded test software, which often comprises a different test 25 application environment than that used by the external test controller (i.e., different processor architecture, different operating system, and different software drivers). This requires extra work in addition to test development and debug with the external test controller and therefore adds to the 30 total system cost and complexity.

Moreover, in the conventional embedded BIST configuration 400, the microprocessor 402 is often shared or re-used as the test processor. This approach is intended to save costs by not

requiring a separate dedicated processor for use as the embedded test controller. However, the microprocessor 402 and any other support circuitry the microprocessor 402 requires typically cannot be placed in the scan chain(s) of the system during 5 embedded testing because this circuitry is employed to apply and analyze the embedded tests. Accordingly, the fault coverage of the system during embedded testing is reduced because the entire microprocessor infrastructure of the system is not part of the embedded test. Further, a significant portion of the system 10 (e.g., the microprocessor 402 and all its support circuitry) must be free from defects in order to test the remainder of the system. So, although this approach may reduce circuit costs for embedded BIST implementation, test costs may increase. For example, the microprocessor infrastructure of the system may 15 require a separate test methodology and development effort, and may be unable to take advantage of the structured scan methodologies of the remainder of the system.

The need for embedded BIST capabilities in PCBs and systems has grown considerably as the designs of ICs, PCBs, and systems 20 have become more complex. Advances in electronic designs have enabled new product capabilities in the areas of, e.g., telecommunications and information technologies. Such advances have resulted in a growing need for high quality built-in testing and ISC of programmable logic (e.g., CPLDs and FPGAs). 25 Further, increased market demand for such products and increased competition in the market place continue to place pressure on manufacturers of electronic systems to reduce costs and improve time to market. Accordingly, new techniques that both reduce costs and minimize the time required for embedded BIST and ISC 30 of PCBs and systems are needed.

## BRIEF SUMMARY OF THE INVENTION

In accordance with the present invention, an embedded electronic system Built-In Self-Test (BIST) controller architecture is provided that facilitates testing and debugging of electronic circuits and In-System Configuration (ISC) of programmable devices. The presently disclosed system BIST controller architecture reduces costs associated with circuit implementation and scan vector development and debug for system BIST applications. The cost of implementation is reduced because the need for a microprocessor-based system BIST solution is eliminated. In addition, debug time is reduced because the data format of a system BIST controller can be made to correspond to that of an external test controller. As a result, applications debugged with the external test controller can operate successfully in the embedded system BIST controller environment.

Moreover, with the presently disclosed system BIST controller architecture, the engineering time required for custom software development and debug is reduced or eliminated. This includes the time required to run scan vector conversion tools, develop program code using scan function libraries, develop code for the specific microprocessor architecture used, and perform debug in the embedded processor environment.

Further, the system BIST controller architecture comprises a "code-less" solution to system BIST because it does not include a general-purpose microprocessor. Accordingly, the system BIST controller architecture requires less circuitry to implement, thereby reducing both the cost and circuit area needed for the system BIST solution.

The code-less architecture of the system BIST controller has data programmed into a FLASH memory including data for application and analysis of the scan vectors used for embedded system test and ISC. Accordingly, the system BIST controller

architecture may be configured to operate using a Scan Object Format (SOF) compatible with the external test controller.

In one embodiment, the system BIST controller architecture includes the embedded system BIST controller, the embedded FLASH memory circuit, an embedded IEEE 1149.1 bus, and an external controller connector. The system BIST controller is coupled to the FLASH memory circuit and the IEEE 1149.1 bus, and coupleable to the external test controller via the external controller connector. The external test controller can communicate over the IEEE 1149.1 bus (*i.e.*, in a "pass through" mode) to program the FLASH memory and/or the system BIST controller circuitry. In this way, the system BIST controller architecture enables scan vectors to be debugged by the external test controller and then downloaded into the FLASH memory.

Once the scan vectors are programmed into the FLASH memory, the external test controller can be removed or disabled, and the scan vectors can be applied to a circuit or Unit Under Test (UUT) by the system BIST controller in the embedded test environment. This eliminates the need to develop and debug scan vectors for two different environments. As a result, users can easily develop, verify, and debug scan tests using the external test controller before programming them into the FLASH memory.

Because the system BIST controller is configured to apply and analyze scan vectors without requiring a general purpose microprocessor, the system BIST controller can perform on-the-fly comparisons of scan data in hardware. Accordingly, both the data format of the scan vectors and the circuitry of the system BIST controller provide for comparing actual scan-out values, *i.e.*, as output by the circuit or UUT, to expected scan-out values.

The presently disclosed system BIST controller architecture provides for full coverage of the entire functional system. Further, the complete infrastructure of the system processor can

be included in the scan chain(s) for embedded test and can therefore be fully tested by the system BIST controller. Moreover, the system BIST controller architecture enables BIST to be embedded within a system to be tested, on one or more PCBs of the system to be tested, and/or on one or more ICs of the system to be tested, while reducing circuit overhead. In addition, the engineering effort for development and debug of scan vectors is reduced because the system BIST controller architecture provides for a seamless transition from the application of scan vectors using the external test controller to the embedded test application. The system BIST controller architecture does not require a complex microprocessor infrastructure, thereby enabling it to operate independent of the functional processor of the system. Further, the system BIST controller architecture can be implemented either within an existing IC in the system or as a separate dedicated test circuit. As a result, the system BIST controller architecture reduces costs and improves time-to-market.

Other features, functions, and aspects of the invention will be evident from the Detailed Description of the Invention that follows.

#### BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWING

The invention will be more fully understood with reference to the following Detailed Description of the Invention in conjunction with the drawings of which:

Fig. 1 is a block diagram of the conventional IEEE 1149.1 test access port and boundary scan architecture;

Fig. 2 is a block diagram of the conventional IEEE 1149.1 standard bus in a daisy chained configuration;

Fig. 3 is a block diagram of the conventional IEEE 1149.1 standard bus in a multi-drop configuration;

Fig. 4 is a block diagram of a conventional configuration for providing embedded built-in self-test capability in an electronic device or system;

5 Fig. 5 is a block diagram of an electronic system built-in self-test controller architecture according to the present invention;

Fig. 6 is a block diagram of an electronic system built-in self-test controller included in the architecture of Fig. 5;

10 Fig. 7 is a diagram illustrating the format of a scan object employed by the controller of Fig. 6;

Fig. 8 is a diagram illustrating the organization of a FLASH memory included in the architecture of Fig. 5; and

15 Fig. 9 is a flow diagram illustrating a method of testing, programming, or debugging an electronic circuit or system using the architecture of Fig. 5.

#### DETAILED DESCRIPTION OF THE INVENTION

U.S. Provisional Patent Application No. 60/336,586 filed December 4, 2001 is incorporated herein by reference.

20 Fig. 5 depicts an illustrative embodiment of an electronic system Built-In Self-Test (BIST) controller architecture 500, in accordance with the present invention. In the illustrated embodiment, the system BIST controller architecture 500 includes an embedded IEEE 1149.1 bus 512, an embedded system BIST controller 502 coupled to the IEEE 1149.1 bus 512 and a Digital I/O (DIO) bus 520, an embedded memory 504 coupled to the system BIST controller 502, and an external connector 506 coupled to the system BIST controller 502. The external connector 506 is coupleable to an external test controller 507. In a preferred embodiment, the memory 504 comprises one or more FLASH memory devices such a FLASH EPROM or a FLASH EEPROM. It is understood, however, that the memory 504 may alternatively comprise any suitable type of non-volatile storage including a magnetic disk.

Further, the external test controller 507 may comprise any suitable type of Automatic Test Equipment (ATE) such as an In-Circuit Tester (ICT) or a Personal Computer (PC) based boundary scan tool.

5 Fig. 6 depicts an illustrative embodiment of the system BIST controller 502 included in the system BIST controller architecture 500 (see Fig. 5). In the illustrated embodiment, the system BIST controller 502 includes an external-pass-through circuit 602, a results interface 604, a start/stop select circuit 606, a memory interface 608, a Parallel Decode circuit (P\_Decode) 610, a data conversion and compare unit 611 including a Compare (CMP) circuit 612 and a Parallel-To-Serial Conversion (PTSC) circuit 618, an input buffer 614, and a Test Access Port (TAP) generator circuit 620. The structure and operation of the 10 presently disclosed system BIST controller architecture 500 are described below with reference to Figs. 5-6.

15

#### Reset and Synchronization

Fig. 5 depicts two (2) input signals to the system BIST controller 502 that are omitted from Fig. 6 for clarity of illustration. One input signal is RESET\_N, which is used to reset the system BIST controller 502. For example, the RESET\_N input may be provided by the external test controller 507. In the illustrated embodiment, when the RESET\_N signal is asserted logical low, registers and state machines of the system BIST controller 502 are reset to appropriate initial states so that the controller 502 is ready to start applying scan vectors stored in the FLASH memory 504. The other input signal is MSTRCK, which is a master clock signal used to synchronize all 20 activity in the system BIST controller 502. For example, the MSTRCK signal may be provided by an external clock source (not shown) and used to derive the frequency of the TCK signal on the IEEE 1149.1 bus 512.

External Controller Interface

The external-pass-through circuit 602 (see Fig. 6) is configured to enable the selection of either the external test controller 507 or the system BIST controller 502 for subsequent connection to the IEEE 1149.1 bus 512 and the DIO bus 520. Such selection is made using an External Controller Enable (ECE\_N) input provided by the external connector 506 to the external-pass-through circuit 602. In the illustrated embodiment, when the ECE\_N signal is asserted logical low, the system BIST controller 502 is reset and the IEEE 1149.1 bus 512 and the DIO bus 520 are controlled with external TAP\_DIO signals provided by the external test controller 507 via the external connector 506. In the presently disclosed embodiment, connecting the external test controller 507 to the external connector 506 automatically asserts the ECE\_N signal logical low and switches control of the IEEE 1149.1 bus 512 and the DIO bus 520 from the system BIST controller 502 to the external test controller 507. As a result, an External Controller Active (ECA\_N) output signal provided by the external-pass-through circuit 602 goes logical low to indicate that the external test controller is switched to control the IEEE 1149.1 bus 512 and the DIO bus 520. When the ECE\_N signal is logical high, the system BIST controller 502 controls the IEEE 1149.1 bus 512 and the DIO bus 520.

Controlling the IEEE 1149.1 and DIO buses 512 and 520 by the external test controller 507 provides support for test development and debug from the external test controller 507, which in the presently disclosed embodiment includes circuitry compatible with the rest of the system BIST controller architecture 500. For example, the FLASH memory 504 may be programmed by the external test controller 507. In an alternative embodiment, the FLASH memory 504 may be programmed by an external general purpose microprocessor (not shown). In

this case, the external microprocessor controls the DATA bus, the ADDRESS bus, and the CONTROL/STATUS bus of the memory interface 608 included in the system BIST controller 502. By monitoring SBC\_DONE\_N and TEST\_FAIL\_N output signals provided by 5 the results interface 604 and the ECA\_N signal provided by the external-pas-through circuit 602, the external microprocessor can determine whether the DATA bus is free.

#### Test Start/Stop and Select

10 The start/stop select circuit 606 (see Fig. 6) is used to select a scan vector suite to be run by the system BIST controller 502 and start/stop application of the scan vectors. The start/stop select circuit 606 interfaces to the memory interface 608 and has the following inputs:

15 START/STOP: This input causes a START or STOP sequence to occur in the system BIST controller 502. For example, the system's power-on reset circuitry may provide the START/STOP input to the start/stop select circuit 606.

20 Test\_Select: When the START sequence occurs, values on the Test\_Select inputs determine what tests are to be run by the system BIST controller 502. For example, an external switch or jumper may provide the Test\_Select input to the start/stop select circuit 606.

25 In the illustrated embodiment, a rising edge on the START/STOP signal initiates the START sequence in the system BIST controller 502, during which time the values on the Test\_Select inputs are registered and the memory interface 608 is signaled to begin accessing the FLASH memory 504 (see Fig. 5). The Test\_Select values map to a location of the FLASH 30 memory 504 where a predetermined Start Address of the scan vector suite is stored. Once the START sequence is initiated and the system BIST controller 502 begins applying the scan vectors, the START/STOP input to the start/stop select circuit

606 remains logical high to allow the controller 502 to continue applying the selected scan vector sequence. While the system BIST controller 502 is busy applying scan vectors, a falling edge on the START/STOP input causes the system BIST controller 5 502 to halt and subsequently execute a predetermined clean-up sequence. When the START/STOP input is held logical low, the system BIST controller 502 remains in an idle state. In the presently disclosed embodiment, the START/STOP signal is initially pulled-down so that on power-up the system BIST 10 controller 502 remains in its idle state waiting for the first START event.

#### Memory Interface and Organization

The memory interface 608 (see Fig. 6) includes circuitry 15 for communicating with the FLASH memory 504 (see Fig. 5). In the presently disclosed embodiment, the FLASH memory 504 is a word-based FLASH memory. However, it should be understood that the system BIST controller 502 may be configured to interface to other memory types and configurations. The memory interface 608 20 includes inputs and outputs for controlling the FLASH memory 504 as follows.

MRESET\_N: This is an output from the memory interface 608 which when active low, resets the FLASH memory device 504.

CONTROL: This is a set of signals used in controlling 25 erase, program, and read operations of the FLASH memory 504. These signals include Chip Enable (CE), Output Enable (OE), and Write Enable (WE) signals. It is understood that controls vary depending upon the particular FLASH device, manufacturer, and the number of devices used to implement the FLASH memory.

30 STATUS: This is an input to the memory interface 608 for monitoring the Ready/Busy status of the FLASH memory device 504.

ADDRESS: These are outputs from the memory interface 608, which provide the address of a location of the FLASH memory 504 to be read or programmed.

DATA: This is data read from or programmed to the FLASH  
5 memory 504.

Configuration Table

When the system BIST controller 502 (see Figs. 5-6) receives the START signal, the controller 502 causes the memory 10 interface 608 to read a configuration table out of the FLASH memory 504, thereby providing initial configuration and set-up information for the system BIST controller 502. In the presently disclosed embodiment, the configuration table is stored starting at a fixed location assigned to address 0 of the 15 FLASH memory 504. Fig. 8 depicts an exemplary organization 800 of the FLASH memory 504 (see Fig. 5) showing the configuration table 802 stored at the fixed location Config Address.

The configuration table includes the following configuration and timing information:

20 Memory Density: This is an encoded value that indicates the storage size (or depth) of the FLASH memory device 504.

Memory Delay: This is the address-to-output delay time of the FLASH memory device 504, which can be represented as a number of delay cycles.

25 The Memory Density is used by the memory interface 608 for automatically controlling the CE selection. This enables the system BIST controller 502 to support memory configurations implemented with multiple memory banks/devices. In the presently disclosed embodiment, the Memory Density is a pre-determined binary code. For example, an 8-bit binary code of 30 00001010 may indicate that a 256 Mb FLASH memory device is being used, from which the system BIST controller 502 can determine which CE to assert for any selected memory address.

The Memory Delay is used by the memory interface 608 to control the timing of read operations from the FLASH memory 504. Specifically, the Memory Delay is used in generating a time delay, after which read data will be valid and can be accessed 5 from the FLASH memory 504. The Memory Delay is calculated from the FLASH memory's specification for address-to-output delay (e.g., in nanoseconds) and the frequency of the MSTRCK signal (e.g., in MHz). A default Memory Delay is used on power-up reset for initially reading the configuration table from the 10 FLASH memory 504 so that the lowest performance memory device supported by the system BIST controller 502 has sufficient access time to reliably return read data.

Alternative embodiments of the system BIST controller 502 may provide for other configuration and set-up information. The 15 information in the configuration table is specified by a user (e.g., a human operator or a computerized process) and may be programmed into the FLASH memory 504 by the external test controller 507.

20 Selecting a Start Address

After reading the configuration table, the memory interface 608 of the system BIST controller 502 goes to the Start Address location in the FLASH memory 504 determined by the registered Test\_Select inputs, and starts applying the selected scan vector 25 suite. There are  $n + 1$  address locations allocated in the FLASH memory 504 that are mapped to the Test\_Select inputs. Fig. 8 depicts these address locations as a Select Clean-Up Address and Select 1-n Addresses. In the illustrated embodiment, two (2) words comprising high and low segments 806.1 and 806.2 of the 30 Clean-Up address are stored in the FLASH memory 504. Similarly, two (2) words comprising high and low segments 808.1-808.n and 809.1-809.n of the Select 1-n Addresses are stored for each of 16 Select Addresses (e.g., n=15). The Start 1-n Addresses are

programmed into the FLASH memory 504 along with the configuration table and scan vector data.

Once the Start Address is determined, the memory interface 608 begins reading data from the FLASH memory 504 starting at that Start Address and continuing until the end of the scan vectors for this scan vector suite are reached. In the event a failure is detected during the test or there is a falling edge on the START/STOP signal, the memory interface 608 initiates a predetermined clean-up sequence, which is described below.

10

#### Scan Object Organization

Following the Start 1-n Address areas 808.1-808.n and 809.1-809.n of the FLASH memory 504 is a storage area for a plurality of Scan Vector Suites 810. Fig. 8 depicts an exemplary arrangement of the Scan Vector Suites 810 in the FLASH memory 504 (see Fig. 5). Each of the Scan Vector Suites 810 can include any number of tests or ISC data, and each is addressed by a respective Start 1-n Address. For example, the Start 1 Address may comprise the memory location for the start of a first Scan Vector Suite 810. In the presently disclosed embodiment, these scan vectors are applied by the system BIST controller 502 when the Test\_Select inputs to the controller 502 are set to a value of 1 and a START signal is applied to the controller 502.

Fig. 8 depicts Clean-Up Vectors 814 following the Scan Vector Suites 810 in the FLASH memory 504. The Clean-Up Vectors 814 are stored starting at a Clean-Up Address, which is selected as the Start Address in the event the START/STOP input transitions to logical low or a failure is detected during the test. Either of these two events causes the system BIST controller 502 to stop applying the scan vectors, run the Clean-Up Vectors, and then remain idle. In an alternative embodiment, the Clean-Up Address may be selected when the Test\_Select inputs

are logical 0, which in this alternative embodiment may translate to the Select Clean-Up address.

Fig. 8 further depicts an expanded view of the Scan Vector Suites 810 storage area in the FLASH memory 504 showing data stored at Start 1-n Addresses. In the illustrated embodiment, the stored data includes Scan Vector Suites 812.1-812.n followed by respective End/Return commands 822.1-822.n. In the presently disclosed embodiment, the Clean-Up Vectors 814 employ the same scan object format as the Scan Vector Suites 810, including the End/Return command to terminate the vectors.

The system BIST controller 502 provides for re-using a selected set of scan vectors when that set is used in more than one scan vector suite. For example, while formatting scan vectors for subsequent programming into the FLASH memory 504, software in the external test controller may automatically detect and identify any scan vectors that are duplicated in the Scan Vector Suites 810 or the Clean-Up Vectors 814. In the illustrated embodiment, these duplicated scan vectors are called Reused Scan Vectors 816 and are stored in the FLASH memory 504 immediately after the Clean-Up Vectors 814. Fig. 8 depicts the Reused Scan Vectors 816 stored in the FLASH memory 504 starting at Reuse 1 Address.

In the presently disclosed embodiment, the Reused Scan Vectors 816 are stored only once in the FLASH memory 504. Further, a Jump Address command is inserted in place of the reused scan vectors at each instance where these vectors occur in the scan vector suites. For example, the expanded view of the Scan Vector Suites 810 shows a Jump Reuse 1 824 command following the Scan Vector Suite 2 812.2. When the system BIST controller 502 executes the Scan Vector Suite 2 812.2, the Jump Reuse 1 824 command causes the controller 502 to jump to location Reuse 1 Address of the FLASH memory 504, as depicted in an expanded view of the Reused Scan Vectors 816 (see Fig. 8),

thereby allowing Reused Scan Vectors 1 817.1 to be executed by the controller 502. It is noted that Reused Scan Vectors 1 817.1 followed by a Jump Return command 826.1 are stored in the FLASH memory 504 starting at location Reuse 1 Address, and the 5 Reused Scan Vectors 2 followed by a Jump Return command 826.2 are stored in the FLASH memory 504 starting at the location Reuse 2 Address. After the system BIST controller 502 executes the Reused Scan Vectors 1 817.1, the Jump Return 826.1 command causes the controller 502 to return to the memory location where 10 the End/Return 822.2 command is stored, thereby terminating the execution of the scan vectors.

By re-using the Reused Scan Vectors 1-2 817.1-817.2, there is a reduction in the memory storage requirements of the system BIST controller architecture 500 (see Fig. 5). The storage 15 savings corresponds to the size and number of duplicate instances of each re-usable scan object. This can result in maximum utilization of memory storage for the scan vectors.

The Jump Reuse 1 824 command, the End/Return 822.1-822.n commands, and the Jump Return 826.1-826.2 commands provide flow 20 control to the memory interface 608 when applying the Scan Vector Suites 810. For example, these flow control commands may occupy multiple words in the FLASH memory 504, depending on the command type and function. It is noted that whenever a Jump Reuse command is specified, a corresponding Jump Return address 25 is also specified. For example, when the system BIST controller 502 encounters the Jump Reuse 1 824 command when executing the Scan Vector Suite 2 812.2, the controller 502 jumps to the memory location Reuse 1 Address. Next, when the system BIST controller 502 encounters the Jump Return 826.1 command after 30 executing the Reused Scan Vectors 1 817.1 starting at the memory location Reuse 1 Address, the controller 502 uses the Jump Return address as given in the jump command to return from the jump. The End/Return 822.2 command causes the system BIST

controller 502 to stop applying scan vectors and remain in an idle state until another START signal is received. The P\_Decode circuit 610 signals the memory interface 608 that an End/Return command was decoded, thereby causing the memory interface 608 to  
5 stop fetching data from the FLASH memory 504. It is noted that the memory interface 608 also stops fetching data in the event of a test failure, as indicated on a Pass/Fail line by the CMP circuit 612.

10 Conditional Jump

In alternative embodiments of the system BIST controller architecture 500, additional flow control may be provided by conditional jump commands. For example, one such conditional jump command is called an If-Jump Address command. Whereas the  
15 above-described Jump Reuse command directs the system BIST controller 502 to go to a predetermined memory address, the If-Jump Address command directs the controller 502 to go to a predetermined memory address when a compare condition is "true". If the compare condition is "false", then the If-Jump Address  
20 command goes to the address of the next command in the FLASH memory 504. As with the unconditional Jump Address command, when the If-Jump command is executed, the If-Jump command uses the Jump Return command to return to the supplied Jump Return Address.

25 In this alternative embodiment, the If-Jump command comprises a compare of scan data returned from one or more the Units Under Test (UUTs) in the system. Expected scan data is included with the If-Jump command and is compared to actual scan data returned from the UUTs. If the expected and actual data  
30 "compare" (i.e., the compare condition is true), then the If-Jump command transfers the flow of data coming from the memory interface 608 to the specified target address. For example,

conditional jumps may be used to select scan vectors based on UUT type or a particular system configuration.

It is noted that the above-mentioned UUTs are coupleable to the IEEE 1149.1 bus 512 and/or the DIO bus 520 (see Fig. 5).

5 For example, the UUTs may be coupled to the IEEE 1149.1 bus 512 in either the daisy chained or multi-drop configurations 200 and 300 (see Figs. 2-3). Further, the UUTs may be coupled to the DIO bus 520 in any known manner so as to avoid bus contention.

10 Scan Object Format

Fig. 7 depicts an illustrative representation of a portion of a scan object 700, as stored in the FLASH memory 504. It is noted that such scan objects may be stored in the FLASH memory 504 in a binary format. In the presently disclosed embodiment, 15 each of the Scan Vector Suites 810 and the Clean-Up Vectors 814 (see Fig. 8) is formatted as a single scan object. Further, each scan object comprises a plurality of formatted segments such as a Command segment 702, a Reserved segment 704, a Number of Data Bytes High segment 706, a Number of Data Bytes Low 20 segment 708, a System BIST Controller (SBC) Control segment 710, a Cycle Count High segment 712, a Cycle Count Low 714 segment, and a Scan Data segment 716. Data words included in each of these segments are defined as follows.

Command: This segment 702 indicates an SBC operation. 25 Examples of SBC commands include the Scan Vectors command, the End/Return command, the Jump Address command, the If-Jump Address command, the Jump Return command, the Error Code command, and the Message Text command.

Reserved: This segment 704 indicates a word reserved for 30 future use.

# Of Data Bytes High/Low: These segments 706 and 708 comprise two (2) words indicating the size (e.g., the number of data bytes) of the corresponding scan object segment.

SBC Control: This segment 710 provides hardware control settings for selected portions of the system BIST controller 502 (see Figs. 5-6). The control settings are determined and set by user input or automatically by the software of the external test controller 507 (see Fig. 5).

Cycle Count High/Low: Each of these segments 712 and 714 indicates the number of scan clocks to be applied for the command.

Scan Data: For a Scan Vector command, the words of this segment 716 include the actual scan data used for applying and analyzing the scan vectors.

#### Input Buffer and Parallel Decode

As the memory interface 608 reads data words from the FLASH memory 504, the memory interface 608 outputs the data words over an M\_Data bus to the input buffer circuit 614 (see Fig. 6). The input buffer circuit 614 receives the data words over the M\_Data bus, stores the data words, and buffers a number of the data words for processing by the P\_Decode circuit 610.

In the illustrated embodiment, the P\_Decode circuit 610 decodes the data words out of the input buffer circuit 614, generates appropriate controls for other portions of the system BIST controller 502 based on the decoded commands, and sends parallel scan data to the PTSC circuit 618. In the illustrated embodiment, the P\_Decode circuit 610 has the following inputs and outputs:

P\_Control: Based on the decoded commands in the scan objects, the P\_Decode circuit 610 sends control signals to other portions of the system BIST controller 502 over this output bus.

P\_Data: This output bus comprises scan data in parallel word format decoded from scan objects.

Pass/Fail: This input is provided by the CMP circuit 612.

Next\_Addr: This output bus is used to signal a next address to the memory interface 608 for Jump commands.

Done: This output is asserted by the P\_Decode circuit 610 when the circuit 610 finishes applying a set of scan vectors, 5 e.g., when one of the Scan Vector Suites 810 or the Clean-Up Vectors 814 has completed. The system BIST controller 502 goes to the idle state when the Done signal is asserted. This output is applied by the P\_Decode circuit 610 to the memory interface 608 and the results interface 604.

10 P\_Results: This output bus provides decoded results information to the results interface 604. For example, the P\_Results bus may provide extended information (i.e., codes and/or messaging) related to the application of the scan vectors to the results interface 604. The P\_Results information is 15 valid after the P\_Decode circuit 610 has asserted the Done signal.

In the presently disclosed embodiment, the P\_Decode circuit 610 completes the current scan operation when the CMP circuit 612 signals via the Pass/Fail line that a failure has occurred. 20 This assures that the application of the scan data is not interrupted in mid-operation, which may leave partial scan data updated in the system. For example, such partial scan data left in the scan paths of the system that may cause bus contention or some undesirable circuit state that may damage the system or 25 test circuitry before the Clean-Up Vectors are applied.

#### TAP Generator

The TAP generator circuit 620 receives input from the P\_Decode circuit 610 over the P\_Control bus, and generates the 30 IEEE 1149.1 TAP protocol corresponding to the decoded scan objects. The TAP generator 620 generates the TMS, TCK, and TRSTN outputs of the system BIST controller 502 (see Fig. 5). When the ECE\_N signal is logical high, the system BIST

controller 502 sources these signals on the IEEE 1149.1 bus 512. The TAP generator 620 also outputs a control signal S\_Control to the data conversion and compare unit 611.

5    Parallel Conversion and Compare

The data conversion and compare unit 611 (see Fig. 6) includes the CMP and PTSC circuits 612 and 618, which receive input scan data in parallel form from the P\_Decode circuit 610 over the P\_Data bus. The CMP and PTSC circuits 612 and 618 also 10 receive the P\_Control signals from the P\_Decode circuit 610 and the S\_Control signals from the TAP generator 620.

In the presently disclosed embodiment, the PTSC circuit 618 takes test vector data in parallel form, and converts the test vector data into serial form (i.e., scan vector data). For scan 15 test data, the PTSC circuit 618 converts and outputs three (3) streams of serial data - TDO, Mask Data Out (MDO), and Expected Data Out (EDO). TDO is the test data output of the system BIST controller 502 and comprises the source for the TDO of the IEEE 1149.1 bus 512 when the ECE\_N signal is de-asserted. The MDO 20 and EDO signals enable the system BIST controller 502 to analyze the actual scan out data returned from the UUT(s). Accordingly, the MDO and EDO signals, as converted by the PTSC circuit 618, are input to the CMP circuit 612. It is noted that the parallel forms of TDO, MDO, and EDO are stored in the FLASH 25 memory 504 as part of the Scan Data 716 (see Fig. 7).

The CMP circuit 612 receives actual scan-out data from the UUT(s) over TDI, and compares this actual data to the expected scan-out data provided over EDO from the PTSC circuit 618. When the scan data "miscompares" (i.e., the compare condition is 30 "false"), a fault in the UUT has been detected and the CMP circuit 612 signals the P\_Decode circuit 610, the memory interface 608, and the results interface 604 by de-asserting the Pass/Fail signal. By signaling the P\_Decode circuit 610 and the

memory interface 608 via the Pass/Fail line that a failure has occurred, the P\_Decode circuit 610 and the memory interface 608 are directed to run the Clean-Up Vectors.

The PTSC circuit 618 provides the MDO signal to the CMP circuit 612 to allow the system BIST controller 502 to mask one or more of the expected TDI data bits sent back from the UUT(s). For example, such data may be masked when the expected value for a bit of TDI data is specified to be an "X" (i.e., an indeterminate or unknown logic value). Accordingly, when the MDO signal is asserted in the serial data stream, this signal indicates to the CMP circuit 612 that the result of the corresponding TDO-EDO bit compare is to be ignored, in effect forcing the comparison of the bit to pass.

When DIO data is applied to the UUT(s) over the DIO bus 520, the associated P\_Data passes directly from the P\_Decode circuit 610, through the data conversion and compare unit 611, and out over the DIO\_OUT bus. In the illustrated embodiment, the DIO\_OUT bus is selected as the source for the DIO outputs of the system BIST controller 502 when the ECE\_N signal is logical high. The system BIST controller 502 can also receive DIO data from the UUT(s). This DIO data from the UUT may be input to the CMP circuit 612 over the DIO\_IN bus, and compared using the Expected DIO (EDIO) and Mask DIO (MDIO) inputs to the CMP circuit 612. The EDIO and MDIO inputs are provided as parallel data via the P\_Data output of the P\_Decode circuit 610.

#### Results Interface

The results interface 604 reports the outcome of a test or a set of scan vectors, and provides failure and diagnostic information that may be monitored by the user. The results interface 604 has the following inputs and outputs:

TEST\_FAIL\_N: This output is asserted logical low to indicate that a failure has been detected by the CMP circuit 612 during a test.

5 SBC\_DONE\_N: After the system BIST controller 502 is finished running a set of scan vectors, this output is asserted logical low to indicate that the system BIST controller 502 is no longer busy.

Pass/Fail: This input is provided by the CMP circuit 612 to indicate whether the scan vectors passed or failed.

10 Done: This input is provided by the P\_Decode circuit 610 and is asserted when application of the selected scan vectors has completed.

15 P\_Results: The P\_Decode circuit 610 provides extended information related to the application of the scan vectors to the results interface 604 over this input bus. For example, the P\_Results information may comprise an error code or a text message.

20 Pass/Fail\_Code: The results interface 604 decodes the P\_Results information and provides the Pass/Fail\_Code output to the memory interface 608. For example, the Pass/Fail\_Code may be driven out over the DATA bus for diagnostics purposes when a failure has been detected.

25 TXD and RXD: These signals comprise Transmit Data (TXD) and Receive Data (RXD), respectively, of a Universal Asynchronous Receiver/Transmitter (UART) port of the system BIST controller 502.

30 As described above, the results interface 604 receives the Pass/Fail input from the CMP circuit 612 and the Done signal from the P\_Decode circuit 610. In the presently disclosed embodiment, the Pass/Fail signal is asserted logical high after the scan vectors are applied if all bits in the scan test compare successfully. If one or more bits in the scan test do not compare successfully, the Pass/Fail signal goes logical low.

The results interface 604 drives the SBC\_DONE\_N and TEST\_FAIL\_N outputs when the application of the Scan Vector Suite or the Clean-Up Vectors is completed (as indicated by the Done signal).

In the illustrated embodiment, the system BIST controller 5 502 provides for predetermined (e.g., user defined) codes or text messages to be associated with each of the Scan Vector Suites and Clean-up Vectors. These predetermined codes and text messages are passed to the results interface 604 over the P\_Results bus by the P\_Decode circuit 610. The results 10 interface 604 uses the P\_Results data along with the Pass/Fail status and Done status to provide informational messages, pass/fails codes, or diagnostics to the user. The Pass/Fail\_Code is provided by the results interface 604 to the memory interface 608 so that the codes can be driven out over 15 the DATA bus for display purposes. By providing the Pass/Fail\_Code to the DATA bus of the memory interface 608, the code may be displayed by, e.g., an LCD or LED display or read by a system processor connected to the DATA bus. In this case, by monitoring the SBC\_DONE\_N and TEST\_FAIL\_N outputs, an external 20 system processor can determine when the code being output over the DATA bus is valid. This provides flexibility to the user when utilizing the error codes for diagnosis and repair. In addition to the Pass/Fail\_Code, text messages may be provided via the TXD/RXD I/O of the UART port to further aid in diagnosis 25 of failures.

A method of testing, programming, or debugging an electronic circuit or system that includes an embedded BIST circuit according to the present invention is illustrated by reference to Fig. 9. As depicted in step 902, an embedded test 30 bus, an embedded controller, and an embedded memory are provided for the electronic circuit or system. Specifically, the embedded controller is connected to the embedded test bus, and the embedded memory is coupled to the embedded controller.

Further, the embedded memory is communicably coupleable to an external controller. The combination of the embedded test bus, the embedded controller, and the embedded memory makes up the embedded BIST circuit for the electronic circuit or system.

5 Next, data is received, as depicted in step 904, from the external controller by the FLASH memory while the memory is communicably coupled to the external controller. In a preferred embodiment, a "fast access" controller, as described in co-pending U.S. Patent Application No. 09/716,583 filed November, 10 20, 2000 entitled METHOD AND APPARATUS FOR PROVIDING OPTIMIZED ACCESS TO CIRCUITS FOR DEBUG, PROGRAMMING, AND TEST, which is incorporated herein by reference, is employed to program the FLASH memory using the external controller. The received data is subsequently used by the embedded controller for testing, 15 programming, or debugging the electronic circuit or system. The embedded memory also receives predetermined (e.g., user definable) informational data associated with the test data from the external controller. This predetermined informational data can be subsequently communicated to the user by the embedded 20 BIST circuit to aid in diagnosis and/or repair of the electronic circuit or system. It is noted that after step 904 is performed, the external controller may be disconnected from the embedded BIST circuit.

The test data and optionally the informational data are 25 then fetched, as depicted in step 906, from the embedded memory by the embedded controller. Next, the test data is applied, as depicted in step 908, to the electronic circuit or system by the embedded controller for testing, programming, or debugging the electronic circuit or system. Resultant data is then received, 30 as depicted in step 910, from the electronic circuit or system by the embedded controller in response to testing, programming, or debugging the electronic circuit or system. Next, at least a portion of the predetermined informational data is communicated,

as depicted in step 912, to the user by the embedded BIST circuit based on the resultant data to aid the user in diagnosing and/or repairing the electronic circuit or system.

It will further be appreciated by those of ordinary skill in  
5 the art that modifications to and variations of the above-described system BIST controller architecture may be made without departing from the inventive concepts disclosed herein. Accordingly, the invention should not be viewed as limited except as by the scope and spirit of the appended claims.

## CLAIMS

What is claimed is:

1. A method of testing, debugging, or programmably configuring an electronic circuit or system, comprising the steps of:
  - developing and validating data to be applied to the electronic circuit or system by a first test resource, the first test resource being communicably coupleable to the electronic circuit or system via a test bus, the data conforming to a predetermined data format, and the test bus conforming to a predetermined data communications protocol;
  - storing the data in a data store by the first test resource;
  - accessing the data from the data store by a second test resource, the second test resource being communicably coupleable to the electronic circuit or system via the test bus; and
  - applying the data to the electronic circuit or system by the second test resource to test, debug, or programmably configure the electronic circuit or system, the data being applied in the predetermined data format and using the predetermined communications protocol of the test bus.
2. The method of claim 1 wherein the developing step includes identifying common portions of the data, and wherein the storing step includes storing a respective instance of each identified common data portion in the data store.
3. The method of claim 2 wherein the storing step includes storing the respective instances of the identified common data portions contiguously in the data store.
4. The method of claim 2 wherein the developing step further includes developing jump data by the first test resource, the

- jump data corresponding to at least one of the stored instances of the common data portions, and the accessing step includes employing the jump data to access the stored instance of the common data portion from the data store for subsequent 5 application of the common data portion to the electronic circuit or system, thereby enabling multiple non-sequential accesses of the respective common data portions from the data store by the second test resource.
- 10 5. The method of claim 4 wherein the accessing step further includes employing the jump data to access at least one of the stored instances of the common data portions in response to a predetermined condition.
- 15 6. The method of claim 1 further including the step of receiving resultant data from the electronic circuit or system by the second test resource.
- 20 7. The method of claim 6 further including the step of developing informational data based at least in part on the resultant data by the second test resource.
- 25 8. The method of claim 7 wherein the informational data includes at least one predetermined code or text message.
9. The method of claim 6 further including the step of verifying the resultant data by the second test resource.
- 30 10. The method of claim 9 wherein the accessing step includes accessing expected data from the data store, and wherein the verifying step includes comparing the resultant data with the expected data.

11. The method of claim 9 wherein the accessing step includes accessing expected data and mask data from the data store, and wherein the verifying step includes masking at least a portion of the resultant data with the mask data, and comparing the  
5 masked resultant data with the expected data.
12. The method of claim 9 further including the step of, in the event the resultant data is successfully verified, providing an indication of the successful data verification.  
10
13. The method of claim 9 further including the step of, in the event the resultant data fails to be successfully verified, providing an indication of the data verification failure.  
15
14. The method of claim 9 further including the step of, in the event the verifying step is completed, providing an indication of the completion of data verification.
15. The method of claim 1 further including the step of  
20 receiving at least one input signal by the second test resource, the input signal being operative to start or stop the application of the data to the electronic circuit or system in the applying step.
- 25 16. The method of claim 15 wherein the receiving step includes receiving at least one input signal by the second test resource, the input signal being received from a human operator.
- 30 17. The method of claim 15 wherein the receiving step includes receiving at least one input signal in response to at least one event occurring during the application of the data to the electronic circuit or system by the second test resource, the

input signal being operative to stop the application of the data.

18. The method of claim 1 wherein the developing step includes  
5 developing the data to be applied to the electronic circuit or system by the first test resource, the first test resource being communicably coupleable to the electronic circuit or system via the second test resource.

10 19. The method of claim 1 further including the step of sensing a power-up condition by the second test resource, and in the event the power-up condition is sensed, performing the accessing and applying steps.

15 20. The method of claim 1 wherein the developing step further comprises developing a start address table including a plurality of start addresses, each start address being indicative of the start address of a respective partition of the data stored in the data store.

20  
21. The method of claim 20 further including the step of receiving at least one input signal by the second test resource, the input signal being indicative of at least one of the start addresses included in the start address table, and wherein the 25 accessing step includes accessing the data partition corresponding to the start address indicated by the input signal for subsequent application to the electronic circuit or system.

22. The method of claim 1 wherein the developing step includes  
30 developing clean-up data by the first test resource, the clean-up data being operative to place the electronic circuit or system in a predetermined state.

23. The method of claim 22 wherein the applying step includes applying the clean-up data to the electronic circuit or system by the second test resource after completing the testing, debugging, or programmable configuration of the electronic  
5 circuit or system.
24. The method of claim 22 wherein the applying step includes applying the clean-up data to the electronic circuit or system in the event a stop condition is sensed by the second test  
10 resource.
25. The method of claim 22 wherein the applying step includes applying the clean-up data to the electronic circuit or system in the event a predetermined input signal is received by the  
15 second test resource.
26. An apparatus for testing, debugging, or programmably configuring an electronic circuit or system, comprising:  
a test bus communicably connectable to the electronic  
20 circuit or system and to an external test resource, the test bus conforming to a predetermined data communications protocol;  
a resident test resource communicably coupleable to the electronic circuit or system via the test bus; and  
a data store coupled to the resident test resource and  
25 communicably connectable to the external test resource,  
wherein the external test resource is configured to develop and validate data to be applied to the electronic circuit or system, and to store the data in the data store, the data conforming to a predetermined data format, and  
30 wherein the resident test resource is configured to access the data from the data store, and to apply the data to the electronic circuit or system to test, debug, or programmably configure the electronic circuit or system, the data being

applied in the predetermined data format and using the predetermined communications protocol of the test bus.

27. The apparatus of claim 26 wherein the external test  
5 resource is further configured to identify common portions of  
the data, and to store a respective instance of each identified  
common data portion in the data store.

28. The apparatus of claim 27 wherein the external test  
10 resource is further configured to store the respective instances  
of the identified common data portions contiguously in the data  
store.

29. The apparatus of claim 27 wherein the external test  
15 resource is further configured to develop jump data, the jump  
data corresponding to at least one of the stored instances of  
the common data portions, and the resident test resource is  
further configured to employ the jump data to access the stored  
20 instance of the common data portion from the data store for  
subsequent application of the common data portion to the  
electronic circuit or system, thereby enabling multiple non-  
sequential accesses of the respective common data portions from  
the data store by the resident test resource.

25 30. The apparatus of claim 29 wherein the resident test  
resource is further configured to employ the jump data to access  
at least one of the stored instances of the of the common data  
portions in response to a predetermined condition.

30 31. The apparatus of claim 26 wherein the resident test  
resource is further configured to receive resultant data from  
the electronic circuit or system.

32. The apparatus of claim 31 wherein the resident test resource is further configured to develop informational data based at least in part on the resultant data.

5 33. The apparatus of claim 32 wherein the informational data includes at least one predetermined code or text message.

34. The apparatus of claim 31 wherein the resident test resource is further configured to verify the resultant data.

10

35. The apparatus of claim 34 wherein the resident test resource is further configured to access expected data from the data store, and to compare the resultant data with the expected data.

15

36. The apparatus of claim 34 wherein the resident test resource is further configured to access expected data and mask data from the data store, to mask at least a portion of the resultant data with the mask data, and to compare the masked resultant data with the expected data.

20

37. The apparatus of claim 34 wherein the resident test resource is further configured, in the event the resultant data is successfully verified, to provide an indication of the successful data verification.

25

38. The apparatus of claim 34 wherein the resident test resource is further configured, in the event the resultant data fails to be successfully verified, to provide an indication of the data verification failure.

30

39. The apparatus of claim 34 wherein the resident test resource is further configured, in the event the verifying step

is completed, to provide an indication of the completion of data verification.

40. The apparatus of claim 26 wherein the resident test  
5 resource is further configured to receive at least one input signal, the input signal being operative to start or stop the application of the data to the electronic circuit or system.

41. The apparatus of claim 40 wherein the resident test  
10 resource is further configured to receive the input signal from a human operator.

42. The apparatus of claim 40 wherein the resident test  
resource is further configured to receive the input signal  
15 operative to stop the application of the data in response to at least one event occurring during the application of the data to the electronic circuit or system.

43. The apparatus of claim 26 wherein the external test  
20 resource is communicably coupleable to the electronic circuit or system via the resident test resource.

44. The apparatus of claim 26 wherein the resident test  
resource is further configured to sense a power-up condition,  
25 and in the event the power-up condition is sensed, to access the data from the data store, and to apply the data to the electronic circuit or system.

45. The apparatus of claim 26 wherein the communications  
30 protocol of the test bus is the IEEE 1149.1 test standard protocol.

46. The apparatus of claim 26 wherein the test bus comprises a plurality of test buses.

47. The apparatus of claim 46 wherein the plurality of test  
5 buses includes at least one digital test bus.

48. The apparatus of claim 46 wherein the communications protocol of at least one of the plurality of test buses is the IEEE 1149.1 test standard protocol.

10

49. The apparatus of claim 46 wherein the communications protocol of at least one of the plurality of test buses is the IEEE 1149.4 test standard protocol.

15 50. The apparatus of claim 26 wherein the external test resource is further configured to develop a start address table including a plurality of start addresses, each start address being indicative of the start address of a respective partition of the data stored in the data store.

20

51. The apparatus of claim 50 wherein the resident test resource is further configured to receive at least one input signal, the input signal being indicative of at least one of the start addresses included in the start address table, and to  
25 access the data partition corresponding to the start address indicated by the input signal for subsequent application to the electronic circuit or system by the second test resource.

52. The apparatus of claim 26 wherein the external test  
30 resource is further configured to develop clean-up data, the clean-up data being operative to place the electronic circuit or system in a predetermined state.

53. The apparatus of claim 52 wherein the resident test resource is further configured to apply the clean-up data to the electronic circuit or system after completing the testing, debugging, or programmable configuration of the electronic circuit or system.

54. The apparatus of claim 52 wherein the resident test resource is further configured to apply the clean-up data to the electronic circuit or system in the event a stop condition is sensed by the resident test resource.

55. The apparatus of claim 52 wherein the resident test resource is further configured to apply the clean-up data to the electronic circuit or system in the event a predetermined input signal is received by the resident test resource.

56. A method of design for test of an electronic circuit or system, comprising the steps of:

20 providing an external test resource configured to develop and validate data for testing, debugging, or programmably configuring the electronic circuit or system, the data conforming to a predetermined data format;

25 embodying within the electronic circuit or system a test bus conforming to a predetermined communications protocol, the test bus being communicably connectable to the external test resource and the electronic circuit or system, the external test resource being further configured to control the test bus while developing and validating the data;

30 embodying within the electronic circuit or system a data store communicably connectable to the external test resource, the external test resource being further configured to store the data in the data store; and

embodying within the electronic circuit or system a resident test resource, the resident test resource being coupled to the data store and communicably connectable to the test bus, the resident test resource being configured to access the data from the data store, and to control the test bus for applying the data to the electronic circuit or system, the data being applied in the predetermined data format and using the predetermined communications protocol of the test bus.

10 57. The method of claim 56 wherein the providing step includes providing the external test resource, the external test resource being configured to identify common portions of the data, and to store a respective instance of each identified common data portion in the data store.

15 58. The method of claim 57 wherein the external test resource is further configured to store the respective instances of the identified common data portions contiguously in the data store.

20 59. The method of claim 57 wherein the external test resource is further configured to develop jump data, the jump data corresponding to at least one of the stored instances of the common data portions, and the resident test resource is further configured to employ the jump data to access the stored instance 25 of the common data portion from the data store for subsequent application of the common data portion to the electronic circuit or system, thereby enabling multiple non-sequential accesses of the respective common data portions from the data store by the resident test resource.

30 60. The method of claim 59 wherein the resident test resource is further configured to employ the jump data to access at least

one of the stored instances of the common data portions in response to a predetermined condition.

61. The method of claim 56 wherein the third embodying step includes embodying within the electronic circuit or system the resident test resource, the resident test resource being further configured to receive resultant data from the electronic circuit or system.

10 62. The method of claim 61 wherein the resident test resource is further configured to develop informational data based at least in part on the resultant data.

15 63. The method of claim 62 wherein the informational data includes at least one predetermined code or text message.

64. The method of claim 61 wherein the resident test resource is further configured to verify the resultant data.

20 65. The method of claim 64 wherein the resident test resource is further configured to access expected data from the data store, and to compare the resultant data with the expected data.

25 66. The method of claim 64 wherein the resident test resource is further configured to access expected data and mask data from the data store, to mask at least a portion of the resultant data with the mask data, and to compare the masked resultant data with the expected data.

30 67. The method of claim 64 wherein the resident test resource is further configured, in the event the resultant data is successfully verified, to provide an indication of the successful data verification.

68. The method of claim 64 wherein the resident test resource is further configured, in the event the resultant data fails to be successfully verified, to provide an indication of the data  
5 verification failure.

69. The method of claim 64 wherein the resident test resource is further configured, in the event the data verification is completed, to provide an indication of the completion of data  
10 verification.

70. The method of claim 56 wherein the third embodying step includes embodying within the electronic circuit or system the resident test resource, the resident test resource being further  
15 configured to receive at least one input signal, the input signal being operative to start or stop the application of the data to the electronic circuit or system.

71. The method of claim 70 wherein the resident test resource  
20 is further configured to receive at least one input signal, the input signal being received from a human operator.

72. The method of claim 70 wherein the resident test resource is further configured to receive at least one input signal in  
25 response to at least one event occurring during the application of the data to the electronic circuit or system, the input signal being operative to stop the application of the data.

73. The method of claim 56 wherein the providing step includes  
30 providing the external test resource, the external test resource being communicably coupleable to the electronic circuit or system via the resident test resource.

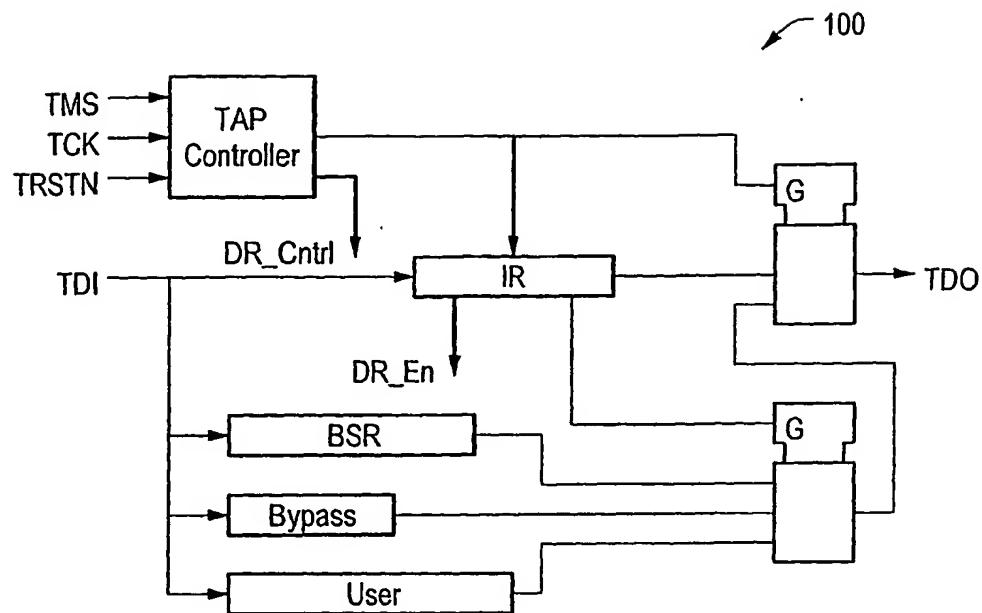
74. The method of claim 56 wherein the third embodying step includes embodying within the electronic circuit or system the resident test resource, the resident test resource being configured to sense a power-up condition, and, in the event the power-up condition is sensed, to access the data from the data store, and to control the test bus for applying the data to the electronic circuit or system.
- 5
75. The method of claim 56 wherein the providing step includes providing the external test resource, the external test resource being configured to develop a start address table including a plurality of start addresses, each start address being indicative of the start address of a respective partition of the data stored in the data store.
- 10
- 15
76. The method of claim 75 wherein the resident test resource is further configured to receive at least one input signal, the input signal being indicative of at least one of the start addresses included in the start address table, and to access the data partition corresponding to the start address indicated by the input signal for subsequent application to the electronic circuit or system.
- 20
77. The method of claim 56 wherein the providing step includes providing the external test resource, the external test resource being configured to develop clean-up data, the clean-up data being operative to place the electronic circuit or system in a predetermined state.
- 25
- 30
78. The method of claim 77 wherein the resident test resource is further configured to apply the clean-up data to the electronic circuit or system after completing the testing,

debugging, or programmable configuration of the electronic circuit or system.

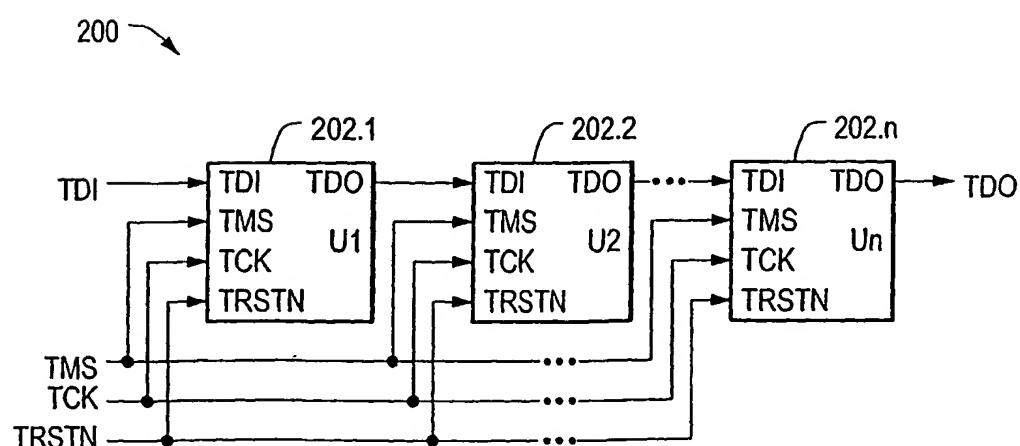
79. The method of claim 77 wherein the resident test resource  
5 is further configured to apply the clean-up data to the electronic circuit or system in the event a stop condition is sensed by the resident test resource.

80. The method of claim 77 wherein the resident test resource  
10 is further configured to apply the clean-up data to the electronic circuit or system in the event a predetermined input signal is received by the resident test resource.

1/7

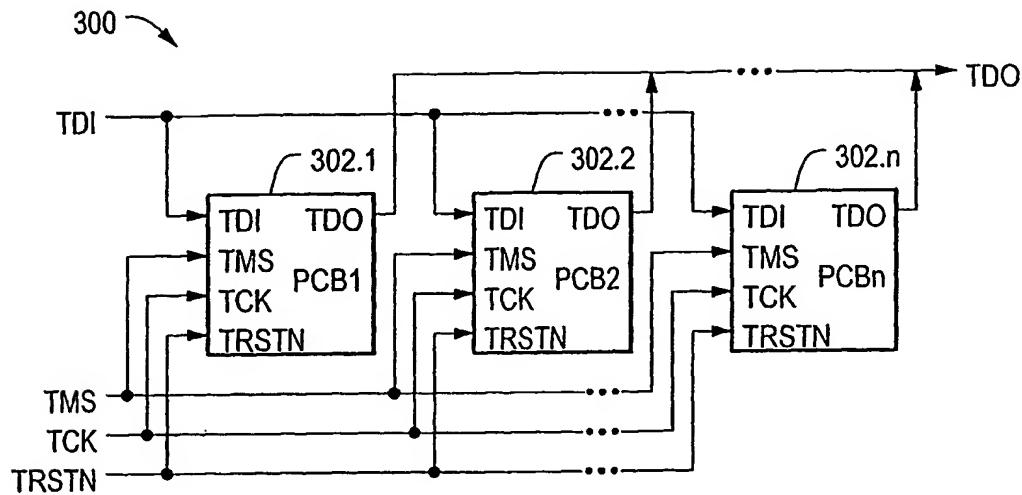


**FIG. 1**  
Prior Art

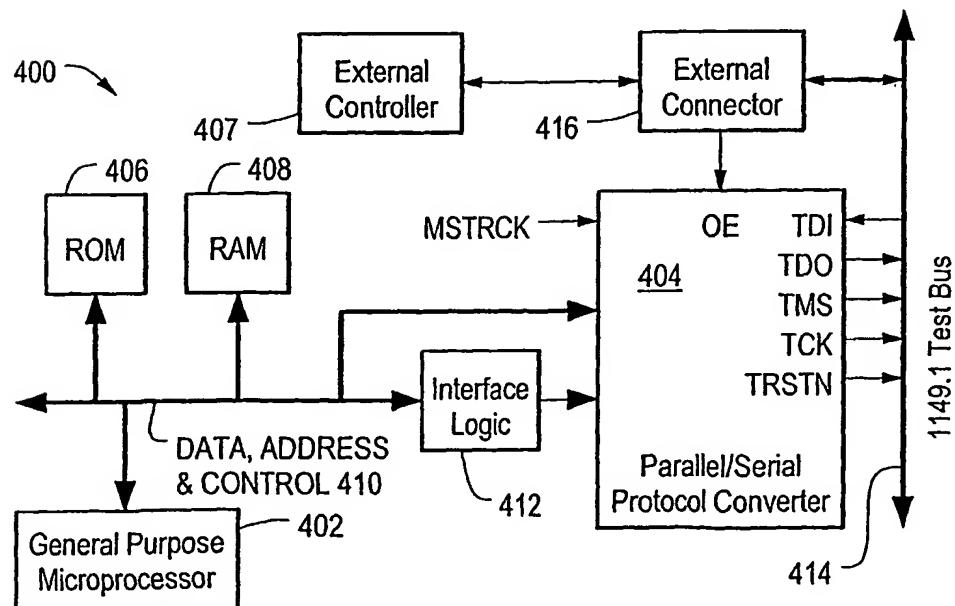


**FIG. 2**  
Prior Art

2/7

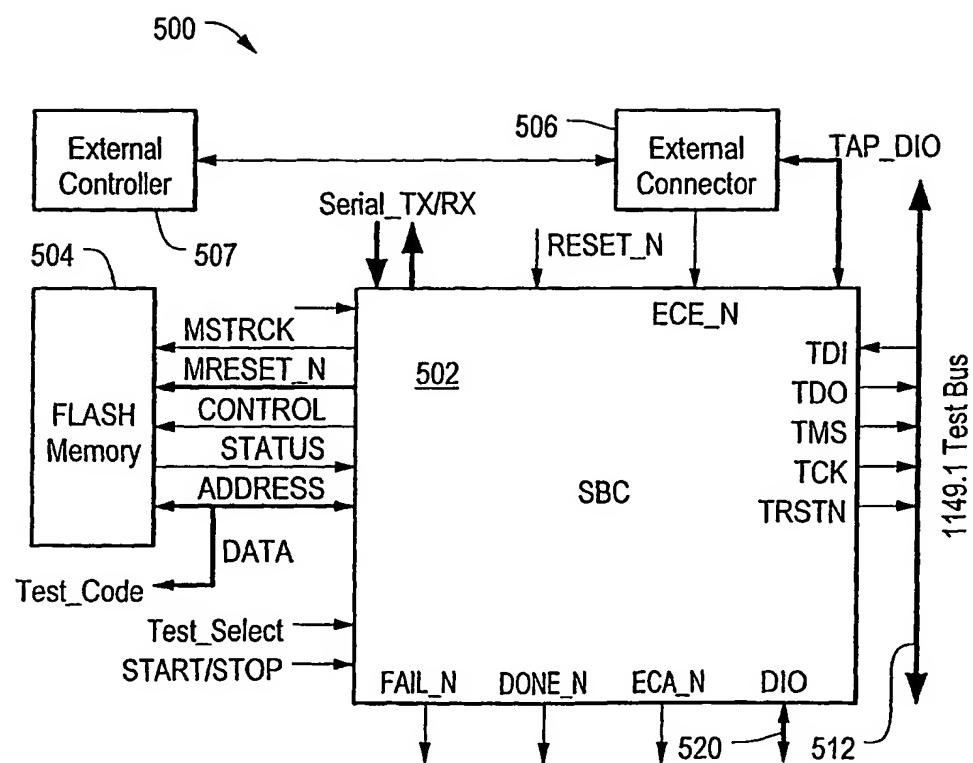


**FIG. 3**  
Prior Art



**FIG. 4**  
Prior Art

3/7

**FIG. 5**

4/7

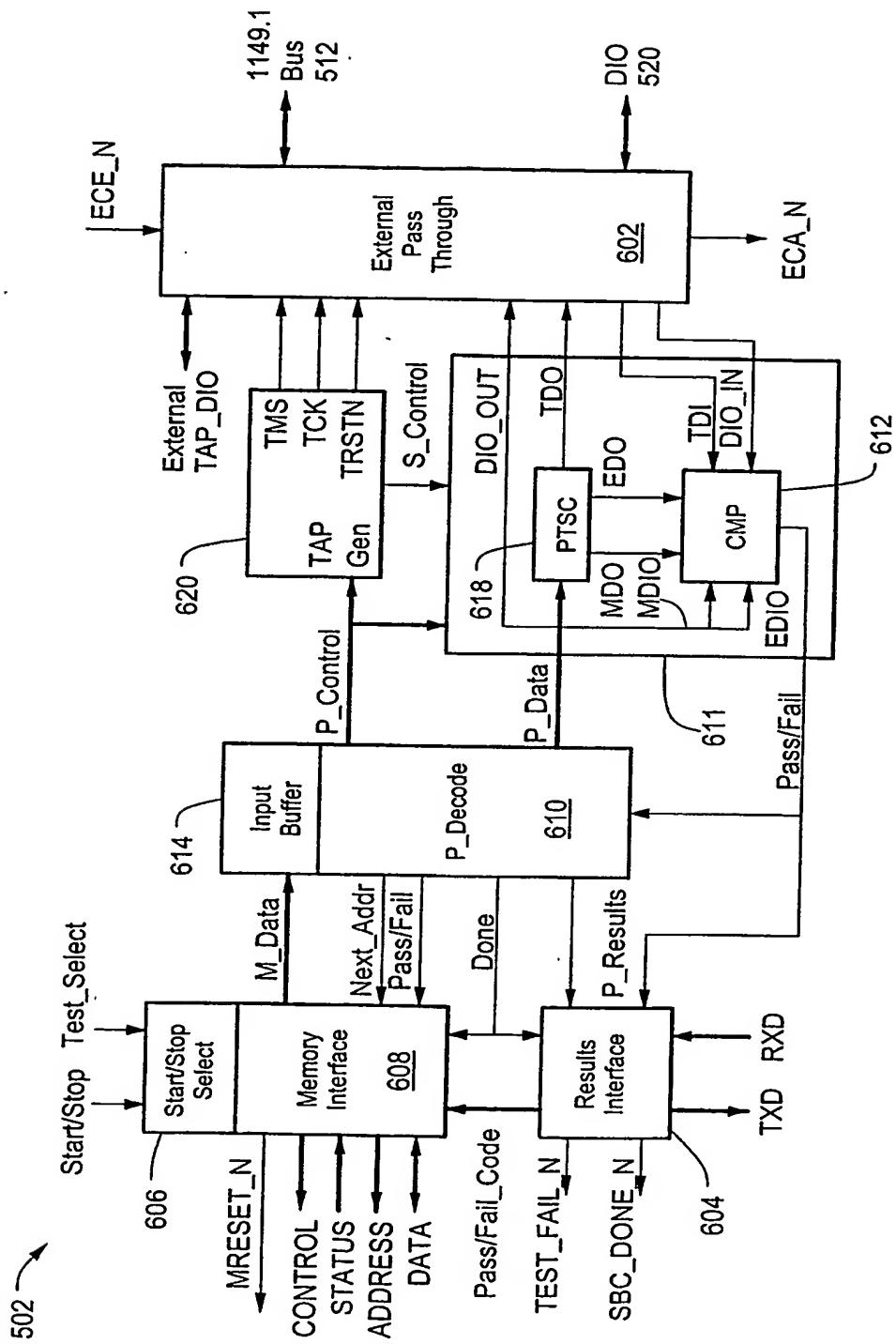


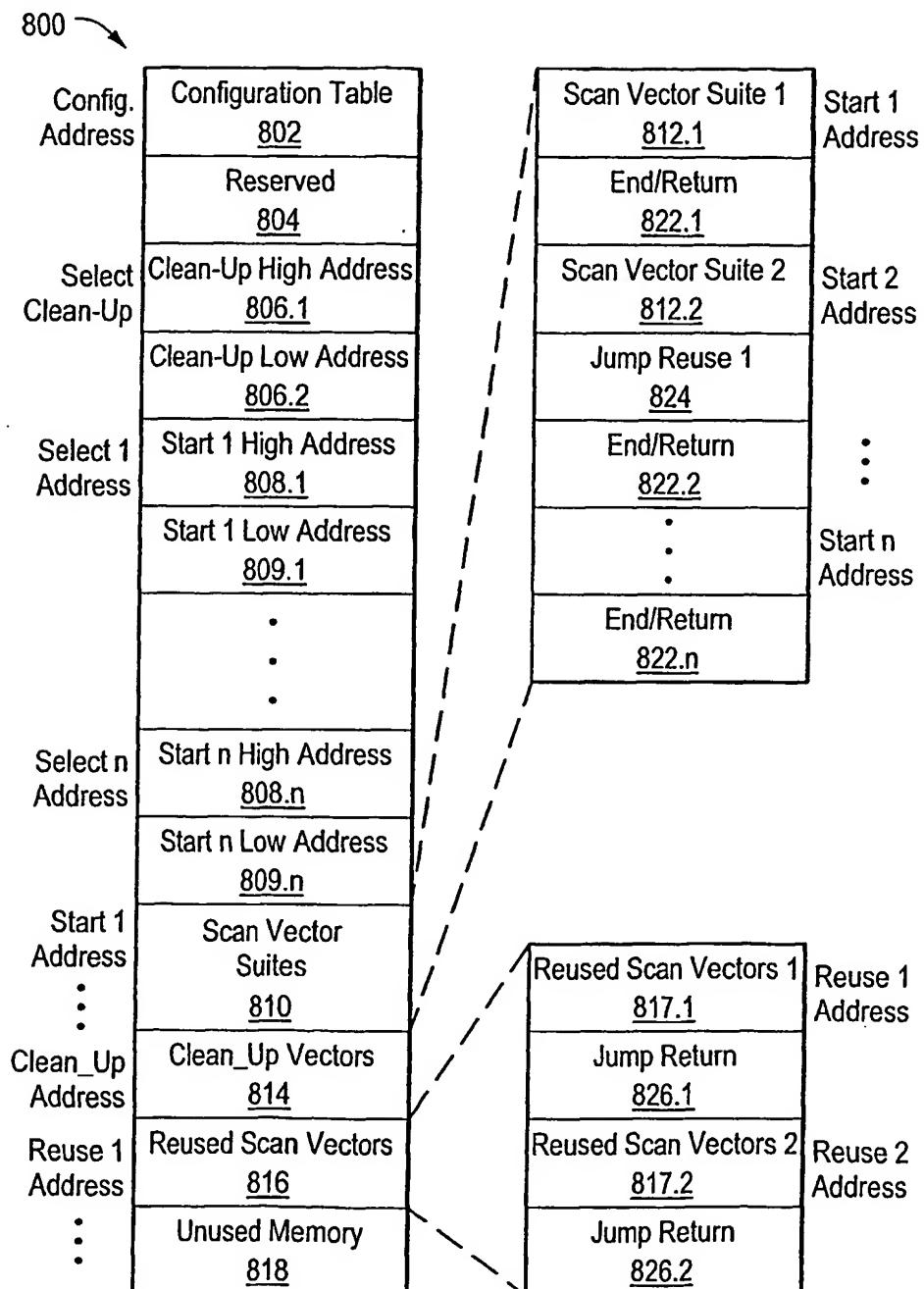
FIG. 6

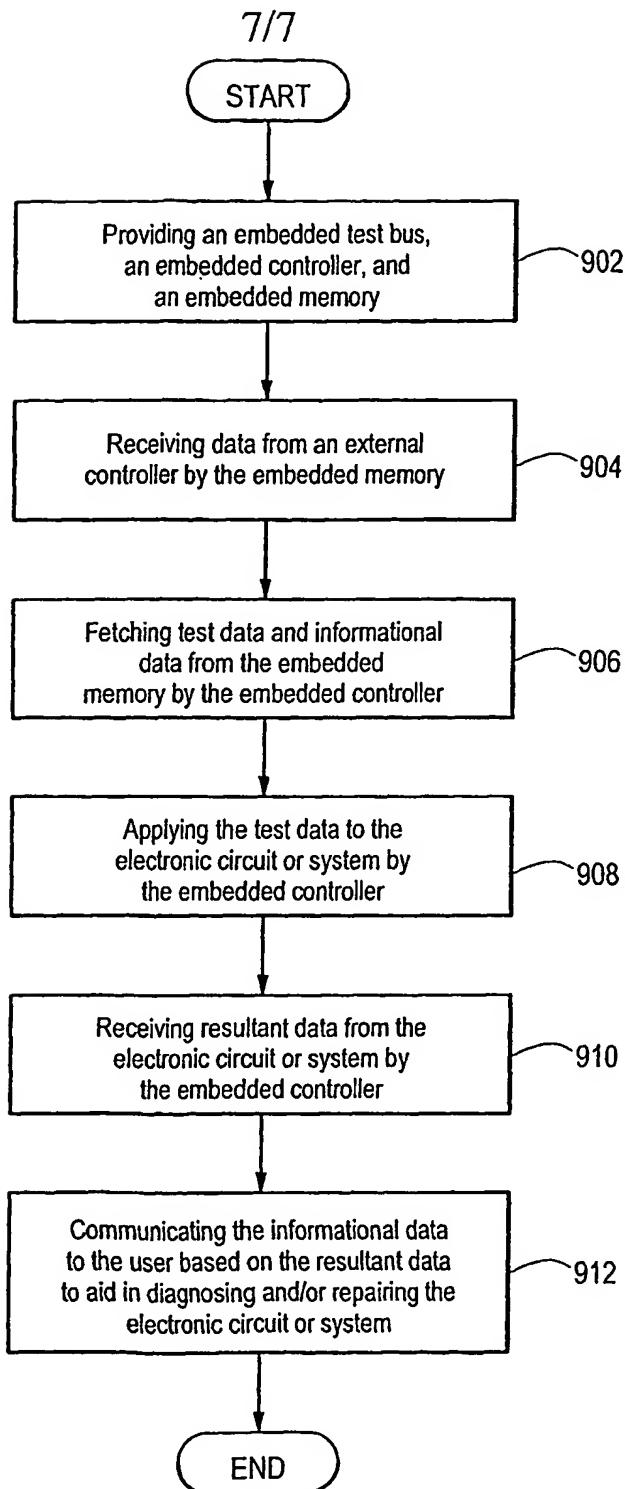
5/7

700	Command <u>702</u>
	Reserved <u>704</u>
	# of Data Bytes High <u>706</u>
	# of Data Bytes Low <u>708</u>
	SBC Control <u>710</u>
	Cycle Count High <u>712</u>
	Cycle Count Low <u>714</u>
	Scan Data <u>716</u>
	•
	•
	•
	•

*FIG. 7*

6/7

*FIG. 8*



**FIG. 9**

## INTERNATIONAL SEARCH REPORT

International application No.

PCT/US02/36246

**A. CLASSIFICATION OF SUBJECT MATTER**

IPC(7) : G01R 31/28  
 US CL : 714/724, 726

According to International Patent Classification (IPC) or to both national classification and IPC

**B. FIELDS SEARCHED**

Minimum documentation searched (classification system followed by classification symbols)  
 U.S. : 714/724, 726; 340/310 R, 723; 324/539, 158 R; 365/185.22; 371/21, 16.2, 5.1

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)  
 IEEE Xplore

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X,P	US 6,427,216 B1 (El-Kik et al.) 30 July 2002 (30.07.2002) claim 1: figure 1, column 3, lines 40-48, column 5, lines 23-28 claim 6: column 4, lines 42-44 claim 9, 10: column 7, lines 31-33 claim 15: column 7, lines 21-22 claim 18: figure 1, column 4, lines 1-7, 12-15, 26-29 claim 22: column 6, lines 41-42 claim 25: column 6, lines 34-42 claim 26: figure 1, column 3, lines 40-48, column 5, lines 23-28 claim 31: column 4, lines 42-44 claims 34-35: column 7, lines 31-33	1, 6, 9, 10, 15, 18, 22, 25, 26, 31, 34, 35

<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C.	<input type="checkbox"/>	See patent family annex.
* Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A" document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E" earlier application or patent published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L" document which may throw doubt on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O" document referring to an oral disclosure, use, exhibition or other means		
"P" document published prior to the international filing date but later than the priority date claimed		
Date of the actual completion of the international search	Date of mailing of the international search report	
31 December 2002 (31.12.2002)	15 JAN 2003	
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703)305-3230	Authorized officer Albert Decady Telephone No. (703)305-3900	

## INTERNATIONAL SEARCH REPORT

## C. (Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X,P	US 6,427,216 B1 (El-Kik et al.) 30 July 2002 claim 40: column 7, lines 21-22 claim 43: figure 1, column 4, lines 1-7, lines 12-15 claim 45: figure 1, column 1, lines 56-62, column 4, lines 16-18 claim 46: figure 1 claim 47: column 3, lines 42-45 claim 48: figure 1, column 1, lines 56-62, column 4, lines 16-18 claim 52: column 6, lines 41-42 claim 55: figure 2, column 6, lines 38-42 claim 56: figure 1, column 3, lines 40-48, column 5, lines 23-28 claim 61: column 4, lines 42-44	40, 43, 45-48, 52, 55, 56, 61
X,P	US 6,427,216 B1 (El-Kik et al.) 30 July 2002 (30.07.2002) claims 64-65: column 7, lines 31-33	64, 65, 70, 73, 77, 80
Y,P	claim 70: column 7, lines 21-22 claim 73: figure 1, column 4, lines 1-7, 12-15 claim 77: column 6, lines 41-42 claim 80: column 6, lines 34-42	7, 32, 62
	claims 7, 32, 62: col. 4, lines 42-44	
Y	US 5,355,493 A (Silberbauer et al.) 11 Oct. 1994 (11.10.1994) claims 2-3, 4-5, 27-28, 29-30, 57-58, 59-60: column 15, lines 45-47	2-3, 4-5, 27-28, 29-30, 57-58, 59-60
Y	US 4,989,207 (Polstra) 29 Jan. 1991 (29.01.1991) claims 4-5, 29-30, 59-60: figure 7, column 9, lines 23-25	4-5, 29-30, 59-60
Y	US 5,799,023 A (Tanaka et al.) 25 Aug. 1998 (25.08.1998) claims 7-8, 32-33, 62-63: column 13, lines 60-61	7-8, 32-33, 62-63
Y	US 4,369,511 (Kimura et al.) 18 Jan. 1983 (18.01.1983) claims 11, 36, 66: figure 2, column 4, lines 43-55	11, 36, 66
Y	US 5,675,540 A (Roohparvar) 7 Oct. 1997 (07.10.1997) claims 12-13, 37-38, 67-68: column 9, lines 34-38 claim 14, 39, 69: column 9, lines 32-34	12-14, 37-39, 67-69
Y	US 6,154,855 A (Norman) 28 Nov. 2000 (28.11.2000) claims 16, 41, 71: column 41, lines 6-8	16, 41, 71
Y	US 5,553,063 A (Dickson) 3 Sep. 1996 (03.09.1996) claims 17, 42, 72: column 4, lines 17-20	17, 42, 72
Y	US 5,051,720 (Kittirutsunetorn) 24 Sep. 1991 (24.09.1991) claims 19, 44, 74: column 6, lines 21-27	19, 44, 74
Y	US 4,742,344 (Nakagawa et al.) 3 May 1988 (03.05.1988) claims 20, 50, 75: figure 2, column 2, lines 29-31 claims 21, 51, 76: column 6, lines 36-42	20, 21, 50, 51, 75, 76
Y	US 5,144,230 A (Katoozi et al.) 1 Sep. 1992 (01.09.1992) claims 23, 53, 78: figure 1, column 7, lines 11-15	23, 53, 78
Y	US 5,638,004 A (Combs et al.) 10 June 1997 (10.06.1997) claims 24, 54, 79: column 14, lines 44-46	24, 54, 79

## INTERNATIONAL SEARCH REPORT

PCT/US02/36246

## C. (Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	Kac, U.; Novak, F.; Macek, S.; Zarnik, M.S.; Alternative test methods using IEEE 1149.4, Design, Automation and Test in Europe Conference and Exhibition 2000. Page(s): 463-467, especially page 463 claim: 49	49

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record.**

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- BLACK BORDERS**
- IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- FADED TEXT OR DRAWING**
- BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- SKEWED/SLANTED IMAGES**
- COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- GRAY SCALE DOCUMENTS**
- LINES OR MARKS ON ORIGINAL DOCUMENT**
- REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- OTHER:** \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**